CLAIMS

What is claimed is:

1. A system for processing graphics data in a computer system comprising:

a general-purpose processor including an instruction queue, the instruction queue for holding a plurality of instructions;

a vector co-processor, coupled with the general-purpose processor, for processing at least a portion of the graphics data using a portion of the plurality of instructions, the vector co-processor capable of performing a plurality of mathematical operations in parallel, the plurality of instructions being written in a general-purpose programming language.

2. The system of claim 1 wherein the plurality of mathematical operations are a plurality of multiply operations and wherein the vector co-processor further includes:

a plurality of multipliers for performing the plurality of multiply operations in parallel.

3. The system of claim 2 wherein the plurality of multipliers provide a first plurality of resultants and wherein the vector co-processor further includes:

an adder tree, coupled to the plurality of multipliers, including at least one stage, the adder tree for adding at least a portion of the first plurality of resultants;

a plurality of resultant paths coupled with the plurality of multipliers and the adder tree for providing the plurality of resultants from the plurality of multipliers or a second plurality of resultants from the at least one stage of the adder tree.

4. The system of claim 1 wherein the vector co-processor further includes:

a first vector buffer and a second vector buffer, wherein the vector co-processor can perform the plurality of mathematical operations on a first portion of the graphics data from the first vector buffer while a second portion of the graphics data is being provided to the second vector buffer and wherein the vector co-processor can perform the plurality of mathematical operations on the second portion of the graphics data from the second vector buffer while the first portion of the graphics data is being provided to the first vector buffer.

5. The system of claim 1 wherein the vector co-processor further includes:

a direct memory access unit for obtaining data directly from a memory.

6. The system of claim 5 wherein the memory includes a system memory.

7. The system of claim 5 wherein the memory includes a local memory.

8. The system of claim 1 wherein the vector co-processor is capable of processing the at least the portion of the graphics data while the general-purpose processor performs a plurality of other operations.

9. The system of claim 8 wherein the plurality of other operations further include processing a second portion of the graphics data.

10. The system of claim 9 wherein the plurality of other operations are used in

processing the graphics data.

11. The system of claim 1 wherein the plurality of operations are a plurality of multiplies, wherein the system further includes a memory and wherein vector co-processor further includes:

a first vector buffer and a second vector buffer, wherein the vector co-processor can perform the plurality of mathematical operations on a first portion of the graphics data from the first vector buffer while a second portion of the graphics data is being provided to the second vector buffer and wherein the vector co-processor can perform the plurality of mathematical operations on the second portion of the graphics data from the second vector buffer while the first portion of the graphics data is being provided to the first vector buffer;

a plurality of multipliers, coupled with the first vector buffer and the second vector buffer, for performing the plurality of multiply operations in parallel to provide a first plurality of resultants;

an adder tree, coupled to the plurality of multipliers, including at least one stage, the adder tree for adding at least a portion of the first plurality of resultants;

a plurality of resultant paths, coupled to the plurality of multipliers and the adder tree, for providing the plurality of resultants from the plurality of multipliers or a second plurality of resultants from the at least one stage of the adder tree;

a resultant mask coupled with the plurality of resultant paths, the first vector buffer and the second vector buffer; and

a direct memory access unit, coupled with the first buffer and the second buffer, for accessing a memory.

12. The system of claim 11 wherein the vector co-processor further includes:

a single operand unit, coupled with the first vector buffer and the second vector buffer, for performing a plurality of operations on a single input from the first vector buffer or the second vector buffer.

13. The system of claim 1 wherein the vector co-processor is capable of performing transformations, clipping and a determination of at least one lighting value for the at least the portion of the graphics data.

14. A method for processing graphics data in a computer system comprising the steps of:

(a) providing a plurality of instructions for processing the graphics data to a general-purpose processor including an instruction queue, the instruction queue for holding the plurality of instructions;

(b) processing the graphics data utilizing a vector co-processor coupled with the general-purpose processor, the vector co-processor processing the graphics data using the plurality of instructions, the vector co-processor processing the graphics data by performing a plurality of mathematical operations on a portion of the graphics data in parallel, the plurality of instructions being provided using software written in a general-purpose programming language.

15. The method of claim 14 wherein the plurality of mathematical operations are a plurality of multiply operations, wherein the vector co-processor further includes a

plurality of multipliers coupled in parallel and wherein the processing step (b) further includes the step of:

(b1) processing the portion of the plurality of data by performing the plurality of multiply operations in parallel using the plurality of multipliers.

16. The method of claim 15 wherein the plurality of multipliers provide a first plurality of resultants, wherein the vector co-processor further includes an adder tree including at least one stage coupled to the plurality of multipliers and a plurality of resultant paths coupled to the plurality of multipliers and the at least one stage of the adder tree, and wherein the processing step (b) further includes the step of:

(b2) adding a first portion of the first plurality of resultants utilizing the adder tree if plurality of instructions includes instructions for adding the first portion of the plurality of resultants; and

(b3) providing the plurality of resultants from the plurality of multipliers or a second plurality of resultants from the at least one stage of the adder tree if the plurality of instructions includes instructions for providing the plurality of resultants from the plurality of multipliers or the second plurality of resultants from the at least one stage of the adder tree.

17. The method of claim 14 wherein the vector co-processor further includes a first vector buffer and a second vector buffer and wherein the processing step (b) further includes the steps of:

(b1) providing a portion of the graphics data alternatively to the first vector buffer

and the second vector buffer;

(b2) utilizing the vector co-processor to perform the plurality of mathematical operations on a first portion of the graphics data from the first vector buffer while a second portion of the graphics data is being provided to the second vector buffer and to perform the plurality of mathematical operations on the second portion of the graphics data from the second vector buffer while the first portion of the graphics data is being provided to the first vector buffer.

18. The method of claim 14 wherein the vector co-processor further includes a direct memory access unit and wherein the processing step (b) further includes the step of:

(b1) obtaining a portion of the graphics data directly from a memory of the system utilizing the direct memory access unit.

19. The method of claim 18 wherein the memory includes a system memory.

20. The method of claim 18 wherein the memory includes a local memory.

21. The method of claim 14 wherein the vector co-processor is capable of processing the at least the portion of the graphics data while the general-purpose processor performs a plurality of other operations.

22. The method of claim 21 wherein the plurality of other operations further include processing a second portion of the graphics data.

23. The method of claim 22 wherein the plurality of other operations are used in processing the graphics data.

24. The method of claim 14 wherein the plurality of operations are a plurality of multiplies, wherein the system further includes a memory and wherein vector co-processor further a first vector buffer, a second vector buffer, a plurality of multipliers coupled to the first vector buffer and the second vector buffer, an adder tree including at least one stage coupled with the plurality of multipliers, a plurality of resultant paths coupled to the plurality of multipliers and the adder tree, a resultant mask coupled with the plurality of resultant paths, the first vector buffer and the second vector buffer, a direct memory access unit coupled with the first vector buffer and the second vector buffer, the method including the steps of:

(b1) obtaining a portion of the graphics data directly from a memory of the system utilizing the direct memory access unit

(b2) providing the portion of the graphics data alternatively to the first vector buffer and a the second vector buffer;

(b3) utilizing the multipliers to perform the plurality of multiply operations on a first portion of the graphics data from the first vector buffer while a second portion of the graphics data is being provided to the second vector buffer and to perform the plurality of multiply operations on the second portion of the graphics data from the second vector buffer while the first portion of the graphics data is being provided to the first vector buffer, the plurality of multipliers thereby providing a plurality of resultants;

(b4) adding a first portion of the first plurality of resultants utilizing the adder tree

if plurality of instructions includes instructions for adding the first portion of the plurality of resultants; and

(b4) providing a second portion of the plurality of resultants from the plurality of multipliers or a second plurality of resultants from the at least one stage of the adder tree if the plurality of instructions includes instructions for providing the plurality of resultants from the plurality of multipliers or the second plurality of resultants from the at least one stage of the adder tree

(b5) utilizing the mask to provide the second portion of the plurality of resultants or the second plurality of resultants to the first vector buffer or the second vector buffer if a portion of the plurality of instructions indicate that the second portion of the plurality of resultants or the second plurality of resultants are to be provided to the first vector buffer or the second vector buffer.

25. The method of claim 24 wherein the vector co-processor further includes a single operand unit coupled with the first vector buffer and the second vector buffer, the method further comprising the step of:

(b6) performing a plurality of operations on a single input from the first vector buffer or the second vector buffer if a second portion of the plurality of instructions indicates that the plurality of operations are to be performed on the single input.

26. The method of claim 14 wherein the processing step (b) includes the step of:

(b1) performing a transformation, clipping and a determination of at least one lighting value for the graphics data.